

# COMPUTER CENTRE BULLETIN



## THIS EDITION

This month's Bulletin goes to press with a smile and a chuckle. As you flip over the pages, you will very soon see why!

As well as a little humour to lighten our day, we have the more serious items of information such as the recording of further FORTRAN errors on the PDP 10, and the availability, on request, of a recent publication to Computer Centre clients. More staff news has been gleaned from private resources, and an article on the image of the computer seen through younger eyes, is included. Our second article discussing PACE simulation, is a sequel to the February Bulletin's article *Digital-Analogue Simulation*, both being written by Mr. Leonard Mor. Copies of this Bulletin (Volume 2, Number 2) may be obtained from the Computer Centre.

## STAFF OF THE COMPUTER CENTRE

### OVERSEAS.....

*John Hendry*, post-graduate Diploma in Automatic Computing student and part-time Demonstrator, has been awarded one of the three Nicklin-Macgregor scholarships. They were established by Mr. Charles H. Dyson of New York, to honour the former Premier of Queensland, Sir Francis Nicklin, and his father-in-law, Mr. L.R. Macgregor, the first Director of the Queensland Council of Agriculture and former Australian Ambassador to Brazil. Mr. Macgregor was also the principal figure in the establishment of Queensland's first radio station, 4QG, which began transmission in 1919.

John leaves the Department this month to study for his Doctorate in Chemical Engineering at the University of Wisconsin, U.S.A.

## NEW FACES....

Five students currently enrolled in the postgraduate courses offered by the Department of Computer Science, are working at the Computer Centre as part-time Assistant Systems Programmers. They are *Colin Churchill*, *Don Kinnear*, *Geoff Lang*, *Julie Nicol*, and *David Seddon*.

Under the direction of the Lecturer in Computing (Systems Programming), Chris de Voil, they will undertake general programming work including program conversion and development. With our extreme shortage of staff, their work will be very helpful.

*Mrs. Nancy Findlater* has been appointed Data Preparation Assistant to capably fill the place left by Angela Vidanovic's departure last month from the Computer Centre.

## SCIENCE DISPLAY

Students from the Diplomas in Automatic Computing and Information Processing, post-graduate courses offered by the Department of Computer Science, produced a small demonstration of the capabilities of the PDP 10 at the University of Queensland's Science Display held on June 27th.

Two remote terminals were set up in lecture room B18, about 100 yards from the Centre proper. Mr. John McCray who looks after on-campus television services, set up a remotely-controlled television camera in the machine room, and a manually-operated camera in B18 to televise the proceedings. Pictures were displayed on television monitor screens positioned in B18.

The highlight of the session was the chess-playing competition held during the evening. Three spectators were invited to play chess against the computer continuously, while other demonstration programs were run concurrently for the audience. One contestant conceded defeat in chess, and no decision was gained in the other two matches.

Students also screened films that were lent for the display by the I.C.L.

and I.B.M. Publicity Departments. We wish to express our sincere thanks to these organizations.

### RECENT PUBLICATIONS

The following publication is now available at the Computer Centre:

TECHNICAL MANUAL NO. 1 Supplement C

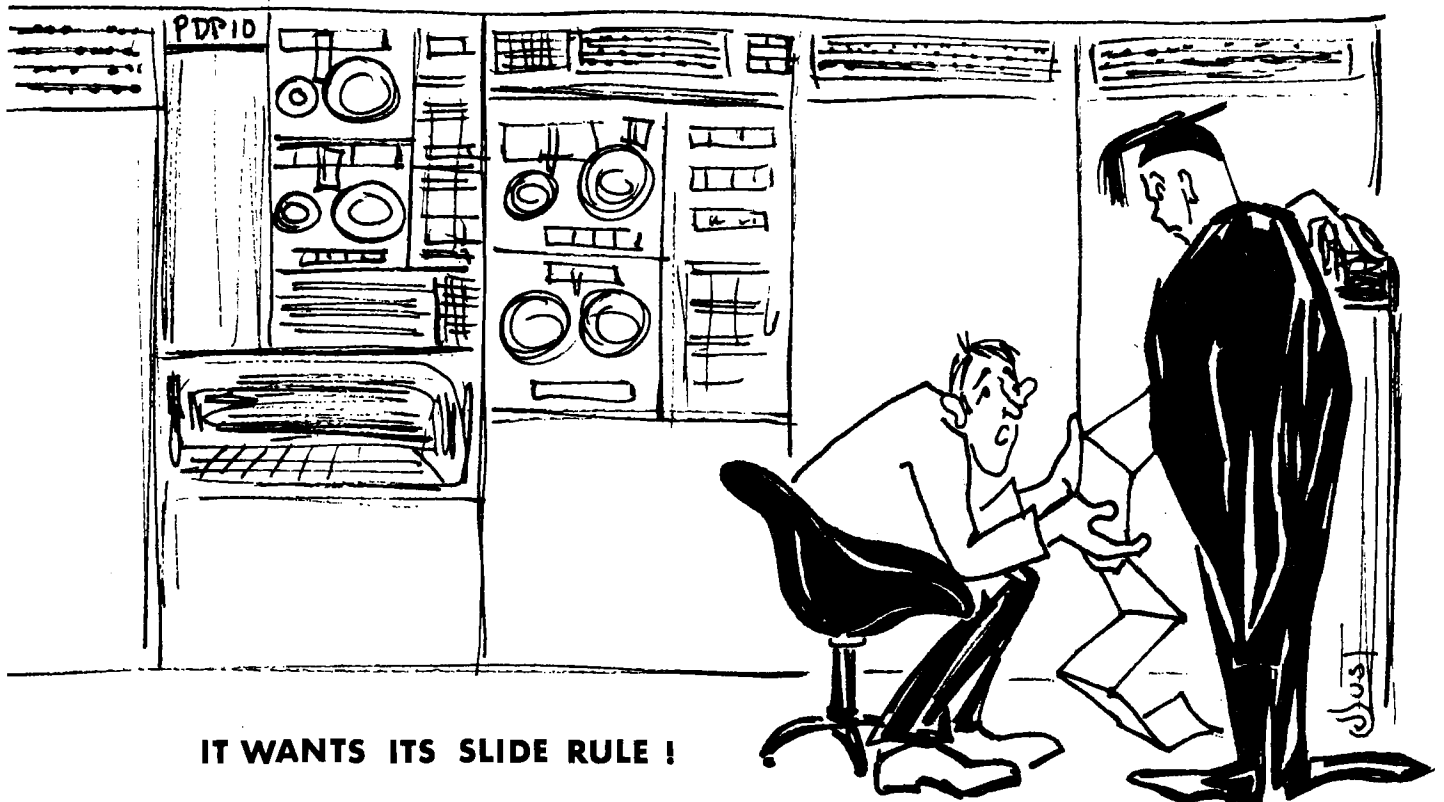
*Converting from GE 225*

*to*

*PDP 10 FORTRAN IV*

J.S. Williams

This supplement is a useful guide for clients who wish to convert programs originally written for the GE 225 for compilation and execution on the PDP 10.



## PDP 10 FORTRAN IV ERRORS

If users feel they have discovered an error, please report it to the Administrative Officer (Mr. John Jauncey, extension 8471) with evidence such as card decks and listings. This will greatly aid members of staff.

1. If a variable in common is used in a subroutine calling sequence, and subsequently, in the subroutine, is defined as being in common, then a conflict of definition occurs. This does not cause a compiler diagnostic message, but may produce unexpected results.

For example:

```
COMMON I
I = 2
J = 1
CALL TEST (I,J)
TYPE 10,I,J
.
.
END
SUBROUTINE TEST (K1,K2)
COMMON K
K2 = 3+K1
K = 4
RETURN
END
```

Diagram illustrating the conflict of definition:

- An arrow points from the `K1` parameter in the subroutine call `CALL TEST (I,J)` to the `K1` parameter in the subroutine definition `SUBROUTINE TEST (K1,K2)`. The text next to this arrow is: "K1 is in fact I, but is called by value."
- An arrow points from the `K` parameter in the subroutine definition `SUBROUTINE TEST (K1,K2)` to the `COMMON K` statement. The text next to this arrow is: "K corresponds to I in the main program."

Effectively, a call-by-value construction is set up for I in the call to subroutine TEST. Thus on return to the main program, the value of K1 last given in the subroutine, will be returned to I. In this example, since K1 was unaltered in the subroutine, the value 2 will be returned, rather than the value of 4 set by the statement `K = 4` in the subroutine.

This is not an actual error, but we offer these examples to indicate the need to take care when using common.

2. Subroutine calls which pass many expressions as arguments, are not compiled correctly because the technique used in evaluation of the expression arguments can overwrite the temporary location used for storing the results of previous argument evaluations. Without extensive further testing, it cannot easily be determined whether this is a function of the number or complexity of sub-expressions, or both.

For example:

```
COMPLEX A,B,C,D
CALL SUB(A*B,A*C,A*D,B*C,B*D,C*D)
```

To overcome the problem, we recommend that you evaluate sub-expressions before making the subroutine call, as in the second example. Thus, compute the expression outside of the argument list:

```
A1 = A*B
A2 = A*C
A3 = A*D
A4 = B*C
A5 = B*D
A6 = C*D

CALL SUB(A1,A2,A3,A4,A5,A6)
```

3. Double-Precision to Single-Precision implicit conversion is not done properly when the number converted is almost a power of 2.

For example:

```
DOUBLE PRECISION D
D = -1073741821D0      (230 = 1073741824D0)
A = D
```

This problem arises because of improper implicit conversion, and it is hoped that Digital Equipment Australia will have remedied this problem in their recently-announced software update.

## LIBRARY ACCESSIONS

This month's list details the books and theses that were added to the shelves of the University Libraries in April 1969.

- McDaniel, Herman. *An Introduction to Decision Logic Tables.* 1968.  
(001.539 MACD, Engin.Lib.)
- Bakewell, Kenneth Graham Bartlett ed. *Library and Information Services for Management.* 1968. (027.69 BAK, Main Lib.)
- Institution of Heating and Ventilating Engineers. *Computation of Heat Transmittance Factors.* 1965. (Q536.2 INS, Architecture Lib.)
- Brazier, Mary Agnes Burniston, ed. *Computer Technique in EEG Analysis.* 1961. (612.82 BRA, Bancroft Lib.)
- Lindberg, Donald A.B. *The Computer and Medical Care.* 1968. (610.18 LIN, Clinical Sciences Lib.)
- Paschkis, Victor. *Direct Analogue Computers.* 1968. (621.381957 PAS, Engin.Lib.)
- Inman, Kenneth L. *Fundamentals of Electronic Data Processing.* 1965. (Q651.8 INM, Architecture Lib.)

## Theses

- |                 |   |
|-----------------|---|
| Pinel, D.B.     | <i>Algorithm for Root Locus Analysis.</i> 1968.<br>PIN, Main Lib. |
| Roubin, B.C.    | <i>Special-purpose Digital Computers.</i> 1968.<br>ROU, Main Lib. |
| Stapinski, T.E. | <i>Error-Correcting Codes.</i> 1968.<br>STA, Main Lib.            |



## COMPUTERS AND THE STUDENT

H.L. Smythe

The post-war period has been characterized by the emergence of the computer as a vital force in almost every aspect of public life. The engineer calculating stresses and strains, the economist predicting patterns of future growth, the psychologist analysing social tension, are only a few of the numerous occupational groups who are finding the computer an invaluable tool in their work. The rise of the computer has been meteoric for it is only a generation old, a brawling, lusty, raucous babe making its presence heard and felt. Its very newness renders it suspect in the public eye, and its capacity for initiating far-reaching change indicates that a computer revolution is in progress.

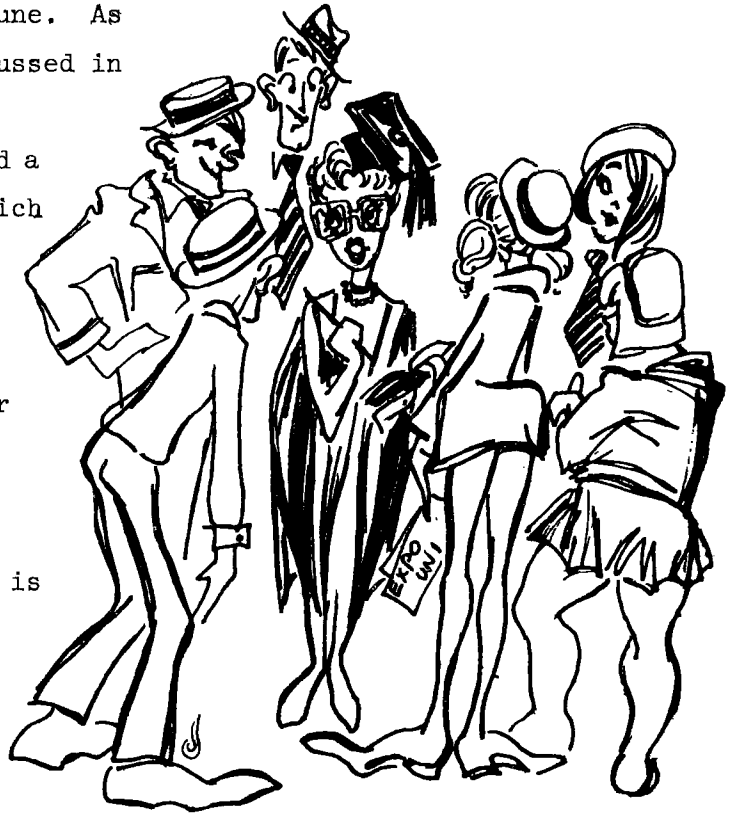
It is little wonder, then, that its public image is a chimera of doubt, suspicion, resentment and fear. The "social" position of the computer is extremely insecure: it needs to be strengthened by the passage of time, hallowed by the laws of usage and acceptance, and sanctified by entry into the shaded cloisters of tradition. The fact that so many people see the computer as a kind of Pandora's Box, ominous with Orwellian overtones of "Big Brother" and "Now Hear This", shows to what extent it is a social outcast! Every age, of course, tends to resist innovations that may radically change the fabric of life. When George Stephenson developed England's first public railway, almost 150 years ago, dire predictions were made that horses would become extinct, and the noise of this mechanical marvel would distract the hens from their serious business of egg-laying! In the same way today, people fear that the computer will displace the working man from his job, and society will be dehumanised and creativity stifled.

To what extent do these misconceptions and fears influence the attitude of the *next* generation of adults towards computers? Do high school students share the common myth that a computer is an electronic brain of almost magic properties? Or do they have a more realistic view of its powers and its limitations? To gauge the overall opinion of school students towards the computer, I conducted an unofficial survey during the annual Science Display

at the University held on the 27th of June. As the role of the computer is rarely discussed in Queensland schools, I was pleasantly surprised to find that most students had a basic understanding and appreciation which was not derived from Scrooge McDuck comics! Far from accepting prevailing prejudices towards electronic wonders, they showed a strong desire to think for themselves and make independent assessments.

The first question raised - what is a computer - elicited the surprisingly thoughtful answer from one boy that it is a "collector, distributor and integrator of facts". Very few compared it to a brain (which **was** a welcome relief), and all students stressed that, as it was **man-made** and man-controlled, it was most unlikely to take over the world! It was interesting to note that many boys were familiar with the concept of programming a series of instructions into the computer, and seemed to realize that the machine does precisely what you tell it - and no more. The basic powers of the computer to add, subtract, multiply and divide were warmly praised by one lad who said that he would like it very much for his homework!

While the students appreciated the amazing rapidity of the computer and its almost unlimited field of applications, they still retained a fairly logical, balanced attitude towards the machine. The consensus of opinion was that the computer should be used as an invaluable aid and supplement to human effort. Suggestions that it could replace humans completely in all kinds of enterprise, were rejected quite vehemently. Students felt that the creativity and originality of human thought were qualities that no computer could ever duplicate. They showed a similar attitude towards the use of medical diagnostic programs. While considering them very useful as an extra aid for the harassed suburban G.P., they felt that the element of calculated guess-work and professional intuition



which is vital in many complicated diagnoses, would be lost. In other words, the human touch and human contact were not merely preferable but necessary.

The majority of the students did not consider the problem of legal control of access to magnetic tapes and memory dumps, urgent or acute. With the typical Australian approach of "she'll be right, mate", they said that the government would automatically legislate for the protection of individuals should the problem assume significant proportions in the future. The minority of students who did appreciate that there was cause for concern, strongly recommended flexibility of control. They felt that a straitjacket of legal rigidity would hinder scientific progress - it would leave no room to move. Control should prevent misuse, abuse, and generally aid effective use of facilities.

Most of the students said that the acceptance of the computer in industry and commerce would not create a wave of unemployment. They maintained that the creation of new jobs associated with the operation of the machine, would absorb many workers, and that employers, in numerous cases, would retrain their existing personnel. Visualizing life in a computer age, they anticipated a lower retiring age, more leisure, and less mundane work which the computer would process with much greater accuracy and infinitely more speed. Far from viewing the computer with suspicion and fear, they had confidence in its powers and optimism in the future developments of science. They agreed that older people had a poor image of the computer and its implications, but entertained no such apprehensions themselves.

My main conclusions from speaking to these students were as follows:

1. They had a fairly logical, practical and sensible attitude towards the "Big Black Box": they realized its limitations, appreciated its powers, and did not view it with superstitious distrust but with youthful optimism.
2. They recognised the need for a balance so that the computer would not duplicate human effort but rather act as an aid and supplement. They felt that human intelligence, inspiration, and creativity were intangible qualities that could not be reproduced electronically.

3. The question of legal control of computers was not deemed urgent. However, some restrictions were advised in the future. Characteristically, they shied away from any authoritarian kind of regulation.
4. Their attitudes in no way reflected the feelings of their parents, e.g. computers bring economic dislocation, or the sensationalism of the press. They have accepted the computer without reservations, naturally, simply, as yet another advancement of science.

### PACE SIMULATION

*Leonard Mor*

In this article, the second on digital simulation (see Volume 2 Number 2, *Digital-Analogue Simulation*), features which have been added to the PACE Simulation language are discussed, and a realistic simulation problem is described.

### STUDY OF CHLORINE RECOVERY BY ELECTROLYSIS

In some organic processes involving chlorination, electrolysis is used to recover chlorine from spent hydrochloric acid. Brennan and Fahidy<sup>1</sup> show that for such a process, the system equations can be reduced to:

---

1. R.D. Brennan and T.Z. Fahidy. *Digital Simulation*. Instruments and Control Systems, pp. 133, March 1966.

$$\begin{aligned}
C' &= 0.287 - 0.000134 I - 0.0287 C & \text{----} & 1.1 \\
I &= (U - E - N_c - N_a)/R & \text{----} & 1.2 \\
E &= 1.358 - 0.0513 \ln(a) & \text{----} & 1.3 \\
N_c &= 0.363 + 0.0847 \ln(I) & \text{----} & 1.4 \\
N_a &= -0.72 + 0.127 \ln(I) & \text{----} & 1.5 \\
R &= 43.2/C/g & \text{----} & 1.6
\end{aligned}$$

where

$C$  = concentration of the acid in the cell (G.E.W./litre)  
 $I$  = current drawn by the cell (amperes)  
 $U$  = supply voltage (volts)  
 $E$  = cell electromotive force (volts)  
 $N_c$  = cathode polarisation overpotential (volts)  
 $N_a$  = anode polarisation overpotential (volts)  
 $R$  = internal resistance of the cell (ohms)  
 $a$  = mean molal activity  
 $g$  = equivalent conductance (mhos/GEW)

The equivalent conductances and mean molal activity are empirically related to the concentration as follows:

| $\sqrt{c}$ | $g$   | $c$   | $a$    |
|------------|-------|-------|--------|
| 0.0        | 426.2 | 0.275 | 0.210  |
| 0.5        | 377.4 | 1.695 | 1.645  |
| 1.0        | 332.3 | 3.475 | 5.980  |
| 1.5        | 270.0 | 5.370 | 20.010 |
| 2.0        | 200.1 | 7.370 | 62.75  |
| 2.5        | 134.7 | 9.460 | 189.0  |
| 3.0        | 83.5  | 11.62 | 570.0  |
| 3.14       | 72.9  |       |        |
| 3.46       | 52.7  |       |        |

TABLE 1. Empirical Relationships of  $c$ ,  $a$ ,  $g$

This study was concerned with optimising the profitability of the recovery process as a function of the supply voltage for an arbitrary cost of \$0.01 /kilowatt hour for the electrical power, and a value of \$0.10 /lb for recovered chlorine.

Here, we are more concerned with the steady-state behaviour of the plant, and an average stand down time of 200 hours was selected. Hence, assuming a chemical conversion rate of 1.05 lbs. of chlorine per amph-hr., the expression for the profit after 200 hours of operation is given by:

$$\text{PROFIT} = U \cdot \text{TI} \cdot 0.00001 - \text{TI} \cdot 0.0105 \dots\dots 1.7$$

where

U = supply voltage

TI = total current used in 200 hours of operation since  
start-up

A flow chart for the above system equations is given in Figure 1. Although it is not immediately obvious from the equations, an implicit functional relationship exists for the variable I (i.e.  $I = f(I)$ ) and there is, therefore, no explicit solution of I. This can be shown more clearly in the flow chart since an (algebraic) loop exists in the upper left hand section of the diagram not containing an integrator. The PREDICTOR element was developed for the solution of such problems, and is indicated as element P1. This element causes an iteration of the solution process until the difference between two successive inputs is less than a (specified) value.

Additional special elements are shown. ALOG and SORT are the logarithmic and square-root function elements available as standard FORTRAN library routines. The curve fitting elements FIT1 and FIT2 use linear interpolation to provide the empirical relations given in Table 1, and are supplied as FORTRAN programs. The above elements are all defined as FUNCTION elements.

It would have been possible to use the REPETITION facility in PACE to find the profit for a number of supply voltages. However, it is now possible to use a form of optimisation control by defining a special 'element' called a SUBROUTINE,

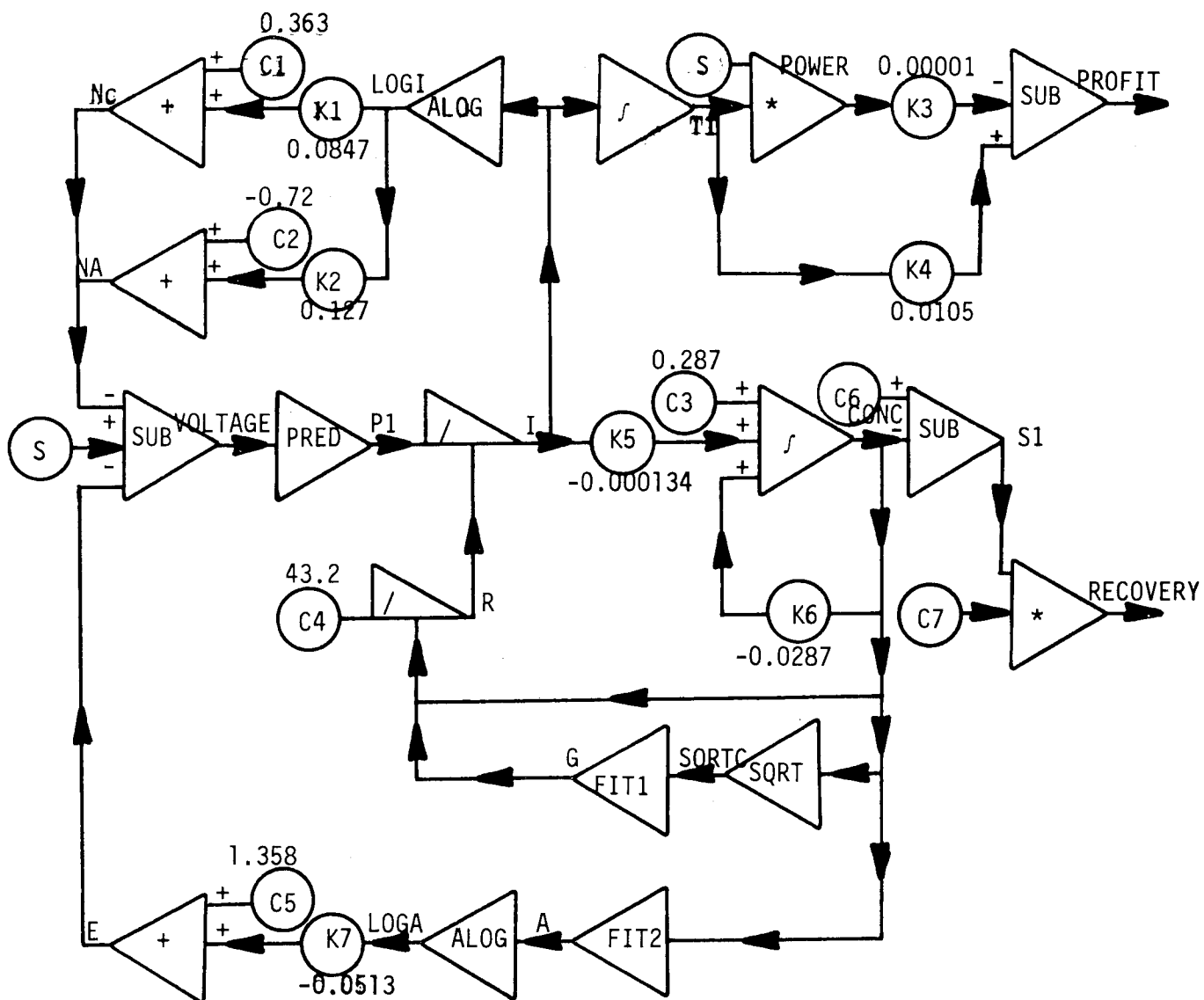


Figure 1. Flowchart for Recovery of Chlorine

and this is demonstrated in the program listing, although not shown on the simulation flow chart.

The control subroutine named CONTROL is also written in FORTRAN IV, and although it is a simple program, requires a deeper understanding of the form of the generated FORTRAN program. In this particular case, the CONTROL routine allows the solution process to continue for 200 'hours', saves this value of profit, resets the value of elements patched to it, increments the supply voltage by a fixed step-size (10 volts), and allows the process to proceed until the profit decreases over two successive sums. At this point, the analogue value TIME is set equal to the solution RUN time and the process is terminated.

Of course, the routine could have been written to modify automatically the supply voltage step-size, and thereby produce an exact solution.

The example illustrates the following points of the user-oriented PACE simulation language:

1. Relatively complex elements can be developed quickly and cheaply as FORTRAN function subroutines - and remember, there is no restriction of the number of a particular type of element.
2. There is no restriction on the number of inputs to an element, although, for elements like the DIVIDER, the order of the patching of input is important.
3. Large systems can be simulated at a very reasonable cost. While this particular problem contained only 36 elements, approximately 400 elements can be accommodated in the version (3).



```

DEFINITION SECTION.
  DEFINE CONSTANTS SUPPLY, STEP, VALUE, C1, C2, C3, C4, C5, C6, C7.
  DEFINE POTENTIOMETERS K1, K2, K3, K4, K5, K6, K7.
  DEFINE INTEGRATORS TI, CONC.
  DEFINE ADDERS NC, NA, E.
  DEFINE SUBROUTINE CONTROL.
  DEFINE FUNCTIONS ALOG LOGI, LOGA.
  DEFINE FUNCTION SORT SQRTC.
  DEFINE FUNCTION FIT1 G.
  DEFINE FUNCTION FIT2 A.
  DEFINE MULTIPLIERS POWER, RECOVERY.
  DEFINE SUBTRACTORS PROFIT($), VOLTAGE, S1.
  DEFINE DIVIDERS R, I.
  DEFINE PREDICTOR P1.

PATCHING SECTION.
  PATCH SUPPLY, VOLTAGE, P1, I, LOGI, K1, NC, VOLTAGE.
  PATCH LOGI, K2, NA, VOLTAGE.
  PATCH C1 TO NC.
  PATCH C2 TO NA.
  PATCH C4, R, I.
  PATCH I, K5, CONC, A, LOGA, K7, E, VOLTAGE.
  PATCH C5 TO E.
  PATCH CONC, SQRTC, G, R.
  PATCH C3, CONC, K6, CONC, R.
  PATCH I, TI, K4, PROFIT, CONTROL.
  PATCH TI, POWER, K3, PROFIT.
  PATCH SUPPLY TO POWER.
  PATCH C6 TO S1.
  PATCH CONC, S1, RECOVERY.
  PATCH C7 TO RECOVERY.
  PATCH VOLTAGE TO CONTROL.
  PATCH SUPPLY TO CONTROL.
  PATCH CONC TO CONTROL.
  PATCH TI TO CONTROL.
  PATCH I TO CONTROL.
  PATCH K5 TO CONTROL.
  PATCH K6 TO CONTROL.
  PATCH STEP TO CONTROL.
  PATCH VALUE TO CONTROL.

DESCRIPTION SECTION.
  HEAD DEMONSTRATION PROBLEM - STUDY OF CHLORINE RECOVERY BY ELECTROLYSIS
  HEAD MAXIMISATION OF PROFIT AS A FUNCTION OF SUPPLY VOLTAGE
  HEAD =====
  PRINT TIME, SUPPLY, VOLTAGE, CONC, RECOVERY, PROFIT.

SPECIFICATION SECTION.
  SET C1= 0.363, C2= -0.72, C3= 0.287, C4= 43.2, C5= 1.358, C6= 10.0,
  C7= 100.0, K1= 0.0847, K2= 0.127, K3= 1.E-5, K4= 0.0105, K5= -1.34E-4,
  K6= -0.0287, -0.287, K7= -0.0513, CONC= 10.0, SUPPLY= 50.0,
  VOLTAGE= 50.0, STEP= 10.0, VALUE= 200.0.
  RUN FOR 200.0 HOURS, INCREMENT BY 1.0 AND OUTPUT EVERY 200.0.

END
0 ERRORS DETECTED

```

.FORTRAN

F40 V007 5-JUL-69 18:35 PAGE 1

```
C SUBROUTINE CONTROL(LCNT,PROFIT,VOLTAG,SUPPLY,CONC,TI,I,K5,K6,STEP, 00010
$VALUE) 00020
C 00030
C OPTIMISATION CONTROL ELEMENT 00040
C 00050
C REAL PROFIT(2),VOLTAG(2),SUPPLY,CONC(2),TI(2),I(2),K5(2),K6(2), 00060
$STEP,VALUE 00070
C INTEGER LCNT,MCNT 00080
C LOGICAL FLAG,FLAG1,FLAG2 00090
C REAL SAVE1,SAVE2,SAVE3,SAVE4,SAVE5,SAVE6,SAVE7,SAVE8 00100
C REAL TIME,RUN,INCREM,OUTPUT,ERROR,TCNT 00110
C COMMON TIME,RUN,INCREM,OUTPUT,ERROR,TCNT,FLAG1,FLAG2 00120
C 00130
C DATA SAVE1,FLAG/0.0,0/ 00140
C 00150
C MCNT=3-LCNT 00160
C IF(FLAG.EQ..TRUE.)GO TO 10 00170
C FLAG=.TRUE. 00180
C SAVE2=SUPPLY 00190
C SAVE3=CONC(LCNT) 00200
C SAVE4=I(LCNT) 00210
C SAVE5=I(LCNT) 00220
C SAVE6=K5(LCNT) 00230
C SAVE7=K6(LCNT) 00240
C SAVE8=4.0*INCREM 00250
C RETURN 00260
C 00270
C 10 IF(TIME.LE.VALUE)RETURN 00280
C IF(PROFIT(MCNT).GE.SAVE1)GO TO 20 00290
C STOP 00300
C 00310
C 20 SAVE1=PROFIT(MCNT) 00320
C PROFIT(MCNT)=0.0 00330
C SUPPLY=SAVE2+STEP 00340
C VOLTAG(LCNT)=SUPPLY 00350
C CONC(MCNT)=SAVE3 00360
C TI(MCNT)=SAVE4 00370
C I(MCNT)=SAVE5 00380
C K5(MCNT)=SAVE6 00390
C K6(MCNT)=SAVE7 00400
C INCREM=SAVE8 00410
C TCNT=VALUE 00420
C TIME=0.0 00430
C FLAG=.FALSE. 00440
C FLAG1=.FALSE. 00450
C FLAG2=.FALSE. 00460
C RETURN 00470
C END 00480
00490
```

GLOBAL DUMMIES

LCNT 201

|        |          |   |
|--------|----------|---|
| PROFIT | 202      |   |
| VOLTAG | 203      |   |
| SUPPLY | 204      |   |
| CONC   | 205      |   |
| TI     | 206      |   |
| I      | 207      |   |
| K5     | 210      |   |
| K6     | 211      |   |
| STEP   | 212      |   |
| VALUE  | 213      |   |
| COMMON |          |   |
| TIME   | /.COMM./ | 0 |
| RUN    | /.COMM./ | 1 |
| INCREM | /.COMM./ | 2 |
| OUTPUT | /.COMM./ | 3 |
| ERROR  | /.COMM./ | 4 |
| TCNT   | /.COMM./ | 5 |
| FLAG1  | /.COMM./ | 6 |
| FLAG2  | /.COMM./ | 7 |

SUBPROGRAMS

EXIT

SCALARS

|        |     |
|--------|-----|
| CONTR0 | 214 |
| SAVE1  | 215 |
| FLAG   | 216 |
| MCNT   | 217 |
| LCNT   | 201 |
| SAVE2  | 220 |
| SUPPLY | 204 |
| SAVE3  | 221 |
| SAVE4  | 222 |
| SAVE5  | 223 |
| SAVE6  | 224 |
| SAVE7  | 225 |
| SAVE8  | 226 |
| INCREM | 2   |
| TIME   | 0   |
| VALUE  | 213 |
| STEP   | 212 |
| TCNT   | 5   |
| FLAG1  | 6   |
| FLAG2  | 7   |
| RUN    | 1   |
| OUTPUT | 3   |
| ERROR  | 4   |

ARRAYS

PROFIT 202

118

081825

F40

V007

5-JUL-69

18:35

PAGE 3

|        |     |
|--------|-----|
| VOLTAG | 203 |
| CONC   | 205 |
| TI     | 206 |
| I      | 207 |
| K5     | 210 |
| K6     | 211 |

CONTRO ERRORS DETECTED: 0

```

C      FUNCTION FIT1(INPUT)                                00010
C      CURVE FITTING ELEMENT 1                             00020
C      REAL INPUT, TABLE1(9), TABLE2(9)                  00030
C      DATA TABLE1, TABLE2/0.0, 0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 3.14, 3.46, 00040
C      $426.2, 377.4, 332.3, 270.0, 200.1, 134.7, 83.5, 72.9, 52.7/ 00050
C      DO 10 I=1, 9                                         00060
C      IF (TABLE1(I).GE.INPUT) GO TO 20                     00070
10    CONTINUE                                              00080
C      STOP                                                  00090
C      FIT1=TABLE2(I-1)+(TABLE2(I)-TABLE2(I-1))*(INPUT-TABLE1(I-1))/ 00100
C      S(TABLE1(I)-TABLE1(I-1))                             00110
C      RETURN                                                00120
C      END                                                    00130
C      STOP                                                  00140
C      FIT1=TABLE2(I-1)+(TABLE2(I)-TABLE2(I-1))*(INPUT-TABLE1(I-1))/ 00150
C      S(TABLE1(I)-TABLE1(I-1))                             00160
C      RETURN                                                00170
C      END                                                    00180
C      STOP                                                  00190
C      STOP                                                  00200

```

GLOBAL DUMMIES

INPUT 60

SUBPROGRAMS

EXIT

SCALARS

FIT1 61

I 62

INPUT 60

ARRAYS

TABLE1 63

TABLE2 74

FIT1 ERRORS DETECTED: 0

F40 V007 5-JUL-69 18:35 PAGE 5

```
C      FUNCTION FIT2(INPUT)                                00010
C      CURVE FITTING ELEMENT 2                             00020
C      REAL INPUT, TABLE1(7), TABLE2(7)                 00030
C      DATA TABLE1, TABLE2/0.275, 1.695, 3.475, 5.370, 7.370, 9.460, 11.62, 00040
C      $0.210, 1.645, 5.980, 20.010, 62.75, 189.0, 570.0/ 00050
C      DO 10 I=1,9                                         00060
C      IF(TABLE1(I).GE.INPUT)GO TO 20                     00070
10    CONTINUE                                           00080
C      STOP                                               00090
C      FIT2=TABLE2(I-1)+(TABLE2(I)-TABLE2(I-1))*(INPUT-TABLE1(I-1))/ 00100
20    $(TABLE1(I)-TABLE1(I-1))                          00110
C      RETURN                                           00120
C      00130
C      00140
C      00150
C      00160
C      00170
C      00180
C      00190
```

F40

V007

5-JUL-69

18:35

PAGE 6

END

00200

GLOBAL DUMMIES

INPUT 60

SUBPROGRAMS

EXIT

SCALARS

FIT2 61

I 62

INPUT 60

ARRAYS

TABLE1 63

TABLE2 72

FIT2 ERRORS DETECTED: 0

11K CORE USED

\*

END OF THIS TASK

.LDR SYS:PACLIB (LIB)

DEMONSTRATION PROBLEM - STUDY OF CHLORINE RECOVERY BY ELECTROLYSIS  
 MAXIMISATION OF PROFIT AS A FUNCTION OF SUPPLY VOLTAGE  
 =====

PAGE 1

| TIME        | SUPPLY      | VOLTAGE     | CONC        | RECOVERY    | PROFIT(\$)  |
|-------------|-------------|-------------|-------------|-------------|-------------|
| 0.00000E+01 | 5.00000E+01 | 5.00000E+01 | 1.00000E+01 | 0.00000E+01 | 0.00000E+01 |
| 2.00000E+02 | 5.00000E+01 | 4.77155E+01 | 5.62442E+00 | 4.37558E+02 | 1.04634E+03 |
| 2.00000E+02 | 6.00000E+01 | 5.76535E+01 | 4.77699E+00 | 5.22301E+02 | 2.20644E+03 |
| 2.00000E+02 | 7.00000E+01 | 6.76084E+01 | 4.10445E+00 | 5.89555E+02 | 2.70608E+03 |
| 2.00000E+02 | 8.00000E+01 | 7.75652E+01 | 3.55844E+00 | 6.44156E+02 | 2.75105E+03 |
| 2.00000E+02 | 9.00000E+01 | 8.75385E+01 | 3.12468E+00 | 6.87532E+02 | 2.74631E+03 |
| 2.00000E+02 | 1.00000E+02 | 9.75189E+01 | 2.77764E+00 | 7.22236E+02 | 3.09943E+03 |
| 2.00000E+02 | 1.10000E+02 | 1.07502E+02 | 2.49520E+00 | 7.50480E+02 | 3.21697E+03 |
| 2.00000E+02 | 1.20000E+02 | 1.17486E+02 | 2.26168E+00 | 7.73832E+02 | 3.31191E+03 |
| 2.00000E+02 | 1.30000E+02 | 1.27473E+02 | 2.07344E+00 | 7.92656E+02 | 3.38118E+03 |
| 2.00000E+02 | 1.40000E+02 | 1.37460E+02 | 1.91289E+00 | 8.08711E+02 | 3.43370E+03 |
| 2.00000E+02 | 1.50000E+02 | 1.47448E+02 | 1.77421E+00 | 8.22579E+02 | 3.47288E+03 |
| 2.00000E+02 | 1.60000E+02 | 1.57438E+02 | 1.65336E+00 | 8.34664E+02 | 3.50107E+03 |
| 2.00000E+02 | 1.70000E+02 | 1.67432E+02 | 1.54718E+00 | 8.45282E+02 | 3.52014E+03 |
| 2.00000E+02 | 1.80000E+02 | 1.77426E+02 | 1.45325E+00 | 8.54675E+02 | 3.53154E+03 |
| 2.00000E+02 | 1.90000E+02 | 1.87421E+02 | 1.36963E+00 | 8.63037E+02 | 3.53642E+03 |
| 2.00000E+02 | 2.00000E+02 | 1.97416E+02 | 1.29474E+00 | 8.70526E+02 | 3.53576E+03 |

EXIT  
+C

.EOJ